

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-34843

(43)公開日 平成9年(1997)2月7日

(51)Int.Cl. ⁸	職別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 15/00	3 9 0	9364-5L	G 0 6 F 15/00	3 9 0
	13/00	3 5 5		13/00
H 0 4 N 7/15		9460-5E	H 0 4 N 7/15	3 5 5

審査請求 未請求 請求項の数15 O L (全 16 頁)

(21)出願番号 特願平7-181690

(22)出願日 平成7年(1995)7月18日

(71)出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72)発明者 深澤 寿彦

東京都大田区下丸子3丁目30番2号キヤノン株式会社内

(72)発明者 岡崎 洋

東京都大田区下丸子3丁目30番2号キヤノン株式会社内

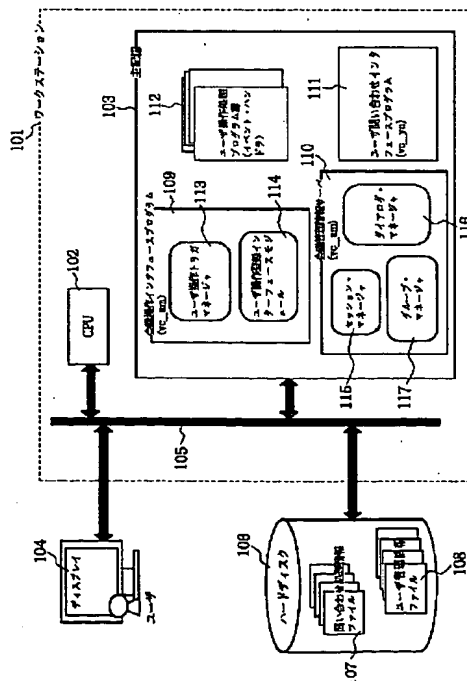
(74)代理人 弁理士 丸島 健一

(54)【発明の名称】 処理システム及び処理装置

(57)【要約】

【課題】 さまざまなタイプの共同作業に対応可能な柔軟な、問い合わせ、また問い合わせに応じた確認を可能とする処理システム及び処理装置を提供することを主な課題とする。

【解決手段】 共同作業の種類を指定するための手段と、前記指定された共同作業において共同作業の対象となる処理装置のユーザまたはユーザグループを指定する手段と、前記指定された共同作業の種類とユーザ、ユーザのグループの組み合わせに対応して問い合わせ処理を指定する手段、前記指定された問い合わせ処理を実行する実行手段とを有する処理装置及び前記共同作業を行う処理装置からなる特徴とする処理システム提供する。



【特許請求の範囲】

【請求項 1】 共同作業の種類を指定するための手段と、
前記指定された共同作業において共同作業の対象となる
処理装置のユーザまたはユーザグループを指定する手段
と、
前記指定された共同作業の種類とユーザ、ユーザのグル
ープの組み合わせに対応して問い合わせ処理を指定する
手段、

前記指定された問い合わせ処理を実行する実行手段とを
有する処理装置及び前記共同作業を行う処理装置から
なる特徴とする処理システム。

【請求項 2】 さらに表示手段を有し、前記実行手段は
少なくとも前記問い合わせ処理に応じた表示を前記表示
手段に表示することを特徴とする請求項 1 項記載の処理
システム。

【請求項 3】 前記共同作業の種類と、前記ユー
ザまたはユーザのグループの指定と、上記問い合わせ処
理を 1 つのファイル上に表として記述することを特徴と
する請求項 1 記載の処理システム。

【請求項 4】 前記問い合わせ処理は通信会議における
問い合わせ処理を含むことを特徴とする請求項 1 記載の
処理システム。

【請求項 5】 前記通信会議における問い合わせ処理
は、会議の開始・終了・参加・退席に対する問い合わせ
を含むことを特徴とする請求項 4 項記載の処理システ
ム。

【請求項 6】 前記ユーザまたはユーザのグループを指
定する手段は、会議の参加者を処理対象として指定する
ことを有することを特徴とする請求項 4 項記載の処理シ
ステム。

【請求項 7】 前記ユーザまたはユーザのグループを指
定する手段は、ある共同作業を要求したユーザを指定す
ることを特徴とする請求項 4 項記載の処理システム。

【請求項 8】 前記ユーザおよびユーザのグループを指
定する手段は、ある共同作業の対象となるユーザ及びユ
ーザのグループを指定することを特徴とする請求項 4 記
項記載のシステム。

【請求項 9】 前記共同作業は、Awareness(状況把握)
における問い合わせ処理を含むことを特徴とする請求
項 1 項記載の処理システム。

【請求項 10】 上記 Awareness(状況把握) における問
い合わせ処理は、ユーザ認証の一部として使用するされ
ることを特徴とする請求項 9 項記載の処理システム。

【請求項 11】 更に共同作業を実行するための、前記
問い合わせを行ったユーザにおける合意の数を指定する
手段を有することを特徴とする請求項 1 記載の処理シ
ステム。

【請求項 12】 前記共同作業を行うためのユーザの合
意の数として、ユーザ具体的な数値を使用せずにグルー

プ内の全ユーザを指定することを特徴とする請求項 1 1
項記載の処理システム。

【請求項 13】 前記共同作業を行うためのユーザの
合意の数として、具体的な数値を使用せずにグループ内
の過半数を指定ことを特徴とする請求項 1 1 項記載の処
理システム。

【請求項 14】 前記問い合わせ処理を、複数のグルー
プに対して段階的に適用することを特徴とする請求項 1
記載の処理システム。

【請求項 15】 複数の処理装置間で共同作業を行う処
理システムにおける処理装置の 1 つであって、
共同作業の種類を指定するための手段と、

前記指定された共同作業において共同作業の対象となる
処理装置のユーザまたはユーザグループを指定する手段
と、

前記指定された共同作業の種類とユーザ、ユーザのグル
ープの組み合わせに対応して問い合わせ処理を指定する
手段、

前記指定された問い合わせ処理を実行する実行手段とを
有することを特徴とする処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は処理システム及び処
理装置に関する。

【0002】特に複数の処理装置間で共同作業を行う処
理システムにおける前記処理装置間の問い合わせ確認機
構に関するものである。

【0003】

【従来の技術】近年、複数のパーソナルコンピュータあ
るいはワークステーション等の処理装置を組み合わせ、
データ交換あるいはデータ共有を可能にすることによ
り、多人数での共同作業をサポートするグループウェア
・システムが提案され、利用されるようになって
きている。

【0004】このグループウェア・システムでは、共同
作業を行うというシステムの性格上 1 人のユーザが行な
った操作が他の複数のユーザのユーザ環境に反映され
る。たとえば、グループウェア・システム上で通信会議
を行っているような場合、1 人のユーザがプレゼンテ
ーション用のアプリケーションを起動すると、参加者全員
のユーザ環境で、プレゼンテーション用のデータが表示
される。

【0005】

【発明が解決しようとする課題】共同作業としては
雑談程度の話し合い

複数人数での会議(議長に相当する参加がいる)

共有データの検討

共有データ上での共同作業

など、さまざまな形式が考えられ、たとえば雑談と正式
な会議では、その開始・終了に必要な問い合わせにおい

て、雑談では特に厳密な問い合わせは必要ないであろうが、正式な会議では議長の最終的な決定がなされなければ開始や終了等を行うべきではなくまた、同じ会議においても、その議題や提案者などの違いによって、問い合わせの方法(この場合は合意の取りかた)は異なってくる。

【0006】このように、共同作業における、ユーザ間の合意および確認をとるために必要な処理は非常に多岐にわたることになる。

【0007】本発明は上述の問題を踏まえてなされた発明であり、さまざまなタイプの共同作業に対応可能な柔軟な、問い合わせ、また問い合わせに応じた確認を可能とする処理システム及び処理装置を提供することを主な課題とする。

【0008】

【課題を解決するための手段】本発明は上述の課題を解決することを目的とし、共同作業の種類を指定するための手段と、前記指定された共同作業において共同作業の対象となる処理装置のユーザまたはユーザグループを指定する手段と、前記指定された共同作業の種類とユーザ、ユーザのグループの組み合わせに対応して問い合わせ処理を指定する手段、前記指定された問い合わせ処理を実行する実行手段とを有する処理装置及び前記共同作業を行う処理装置からなる特徴とする。

【0009】また本発明は、複数の処理装置間で共同作業を行う処理システムにおける処理装置の1つであって、共同作業の種類を指定するための手段と、前記指定された共同作業において共同作業の対象となる処理装置のユーザまたはユーザグループを指定する手段と、前記指定された共同作業の種類とユーザ、ユーザのグループの組み合わせに対応して問い合わせ処理を指定する手段、前記指定された問い合わせ処理を実行する実行手段とを有することを特徴とする。

【0010】

【発明の実施の形態】

(第1の実施例)本発明の一実施例である第1の実施例について図を用いながら説明する。

【0011】図1は、本発明の第1の実施例である会議管理システムの構成図である。

【0012】第1の実施例は共同作業のうち、通信会議の形で運用されるものを例示する。本実施例では、このシステムで運用される会議を"セッション"と呼ぶことにする。

【0013】図1において、ワークステーション(101)は本実施例における処理の実行を行うCPU(102)と、本会議管理システムの基本ソフトウェア群(109, 110, 111, 112)を保持する主記憶(103)を計算機バス(105)で結合することにより構成されている。また、本実施例における処理に必要な永続データを保持するためのハードディスク(106)と、ユーザがシステムの操作をおこなうため

のユーザインタフェースを表示するためのディスプレイ(104)がワークステーション(101)に結合されている。なお、後述の問い合わせ処理、確認処理等における種々の操作は不図示のキーボードや、マウス等のポインティングデバイスからの入力より行われるものである。

【0014】一方ハードディスク(106)中には、問い合わせ処理情報ファイル(107)とユーザ管理情報ファイル(108)が置かれている。問い合わせ処理情報ファイル(107)は、本実施例であって説明される問い合わせ処理を制御するために必要な情報が書かれているファイルである。ユーザ管理情報ファイル(108)は、本実施例で説明される問い合わせ処理の対象となるユーザおよびユーザのグループを管理するために必要な情報が書かれているファイルである。

【0015】上述の問い合わせ処理情報ファイル(107)とユーザ管理情報ファイル(108)内の情報は会議管理情報サーバ(110)によって管理されている。

【0016】以下、会議管理情報サーバ(110)等の本システムの基本ソフトウェアについて説明する。なお本システムでは基本ソフトウェアとして以下の4種類のプログラムが提供されている。

【0017】・会議管理情報サーバ(110)

本システムにおけるセッション、ユーザなどの共有情報を集中管理するサーバである。図1においてはワークステーション(101)上に他のプログラムと一緒に動作しているが、会議管理情報サーバ(110)は1つの通信会議にたいし、1つだけ動作させれば良い。このため実際に本システムを運用するばあい、会議管理情報サーバ(110)用に、専用のワークステーションを用意して、そこで動作させるのが一般的である。(この会議管理情報サーバ(110)専用として使用されるワークステーションをSM_HOSTとよぶ。)

会議管理情報サーバ(110)はvc_smというプログラムとして実装されており以降、会議管理情報サーバ(110)をvc_smと呼ぶことにする。vc_smは内部的に3つの構成要素から成り立っている。

【0018】- セッション・マネージャ(115)

会議に関する情報(会議名、内容、参加者、使用するアプリケーションなど)を管理するモジュールであり、ハッシュテーブルに基づく単純なデータベースとして実装されている。

【0019】- ダイアログ・マネージャ(116)

ユーザ問い合わせ処理に関する情報(問い合わせの対象になるユーザ操作の名前、問い合わせ処理に関わるユーザ名、など)を管理するモジュールであり、このモジュールもハッシュテーブルに基づく単純なデータベースとして実装されている。

【0020】- グループ・マネージャ(117)

ユーザおよびユーザのグループに関する情報(ユーザ名、ユーザの現在の状態、グループのメンバなど)を管

理するモジュールでありこのモジュールもハッシュテーブルに基づく単純なデータベースとして実装されている。このモジュールで管理されるユーザおよびグループは、あらかじめユーザ管理情報ファイル(108)内に登録されているが、その他にもあるセッションの参加者からなる一時的なグループの管理も行う。

【0021】・会議操作インタフェースプログラム(109)

vc_amという名前のプログラムとして実装されている。各ユーザのユーザ環境ごとに1つ起動され、ユーザに会議操作のためのインタフェースを提供する。すなわちvc_am(109)はユーザが行なった操作を解釈し、それに対応するvc_sm(110)上の情報の更新や他のユーザのvc_amとのデータ交換、およびユーザ環境で動作する他のプログラム(111, 112など)の制御をおこなう。vc_am(109)は以下の2つの重要なモジュールを持っている。

【0022】・ユーザ操作トリガマネージャ(113)

ユーザ・インタフェースを介してユーザが行なった操作とそれに対応する処理の対応関係を管理するモジュールである。ユーザ操作が発生すると、その操作をイベントとして受けとり、対応する処理を開始する。本実施例では、ユーザ操作に対応する処理は、vc_am(109)やvc_sm(110)とは独立した別プログラムとして実現されている。このため、ユーザ操作トリガマネージャ(113)はユーザ操作の名前と対応するコマンドの対応表として実装されている。

【0023】・ユーザ操作登録インタフェースモジュール(114)

ユーザ操作トリガマネージャ(113)で処理対象となるユーザ操作と、ユーザ操作トリガマネージャ(113)内におけるユーザ操作につけられた名前との関連を作成するための、インタフェースを提供しているモジュールである。本実施例では、ユーザインタフェースをXウィンドウシステムを利用して作成しているため、ユーザ操作登録インタフェースモジュール(114)は、ユーザのウィンドウ上での操作に対して呼び出されるコールバック関数と、ユーザ操作名の対応表として実現される。vc_am(109)内においては、ユーザ操作登録インタフェースモジュール(114)を用いて以下の操作が登録されている：

Session-Start

Session-Terminate

Session-Join

Session-Disjoin

Session-Make-Join

Session-Make-Disjoin

それぞれ、上から会議の開始、会議の終了、会議への途中参加、会議からの途中退席、会議への参加者の追加、会議の参加者の削除に対応している。

【0024】・ユーザ問い合わせインタフェースプログラム(111)

本実施例におけるユーザ操作の問い合わせ処理におけるユーザインタフェースを提供するプログラムである。vc_am(109)等の他のプログラムからの要求にもとづき、図2に示すようなパネルを問い合わせ先ユーザのディスプレイ上に表示する。

【0025】図2において、201はユーザにYESかNOを問い合わせるための問い合わせパネルである。問い合わせパネル(201)は4つの部品から構成されており、202はユーザに問い合わせ内容を表示するためのメッセージ・フィールド、203はタイムアウト情報フィールド、すなわち、ユーザがYES、NOを決定しなかったばあい、数秒後に自動的にどちらかを答えたことになる旨を表示する。図2では、10秒以内に応答がなければNOを答えたとみなす。のこりのYESボタン(204)およびNOボタン(205)はそれぞれ、ユーザが返答を指定するためのボタンである。また、パネル(206)はユーザに情報を提示し確認をもとめる確認パネルである。

【0026】確認パネルの場合は3つの部品から構成されており、207はユーザに情報を提示するためのメッセージ・フィールド、208はタイムアウト情報フィールド、OKボタン(209)はユーザがメッセージ・フィールド内のメッセージを確認した旨を指定するためのボタンである。タイムアウト情報フィールド(208)では15秒後に自動的にこのパネルがクローズされる旨を表示している。

【0027】どちらのパネル(201, 206)においても、ユーザの応答はパネルの表示要求を発行したプログラムに通知される。

【0028】・ユーザ操作処理プログラム群(112)

vc_am(109)の説明で述べたように、ユーザ操作に対する処理は、vc_am(109)やvc_sm(110)が行なう主にユーザ操作処理プログラム群(112)によって実行される。これらのプログラム群をイベント・ハンドラともよび、1種類のユーザ操作に対して1つずつ存在する。

【0029】では、図3と図5,6,7を用いて基本的なユーザ操作問い合わせの処理について説明する、なお図3において図1と同一の処理を行う箇所には同一の名前を付してある。図5,6,7はそれぞれ会議操作インタフェースプログラム(302)、イベント・ハンドラ(305)、ダイアログ・マネージャ(307)の動作をあらわすフローチャートである。

【0030】図3において、ユーザ(301)が適当な操作を行ったと仮定する。その操作が、ユーザ操作登録インタフェースモジュール(303)によってあらかじめ登録されたユーザ操作(前述のように会議の開始、会議の終了など)ならば(図5, S501)、ユーザ操作トリガマネージャ(304)は、その操作に対応するイベント・ハンドラ(305)を起動する(S502)。起動されたイベント・ハンドラ(305)は、ユーザ操作に対応する処理を行なう前に、vc_sm(306)内のダイアログ・マネージャ(307)に、ユーザ

操作に対する問い合わせ処理が指定されているか問い合わせ(308)を行なう(S601)。この問い合わせ要求(308)には、ユーザ操作登録インタフェースモジュール(303)によってユーザ操作に付けられたユーザ操作名が付加される。

【0031】ダイアログマネージャ(307)は問い合わせ(308)からユーザ操作名をとりだし(S701)、それをキーとして、ハードディスク(309)上の問い合わせ処理情報ファイル(310)から対応する問い合わせ処理が指定されているかをしらべ、問い合わせ処理情報ファイル(310)から指定されたユーザ操作に関する部分を取りだし(S702)した後、問い合わせ要求(308)に対するリプライ・メッセージ(311)として、イベント・ハンドラ(305)に送信する(S703)。

【0032】イベント・ハンドラ(305)は、リプライ・メッセージ(311)をしらべ、問い合わせ処理が指定されていなければ(S602)、本来のユーザ操作処理を直ちに実

Type	Req-User	SS-Other-User	Target-User
Session-Start	N	Y(5):all [Req-User&Other-User(5)] X	
Session-Terminate	N	Y(5):all	X
Session-Join	N	X	X
Session-Disjoin	N	X	X
Session-Make-Join	y	X	X
Session-Make-Disjoin	y	X	X
Tool-Start	N	X	X
Tool-Terminate	N	X	X

先頭にTypeと書かれている行には、問い合わせ処理の対象となるユーザおよびグループが列挙される。ここではReq-User, Target-User, SS-Other-Userが指定されているが、これは

Req-User -- 操作を要求したユーザ
Target-User -- 操作の対象となるユーザ
SS-Other-User -- セッション内における、Req-User, Target-User以外のユーザをあらわしている。これらは、vc_sm 中のセッション・マネージャによってあらかじめ組み込まれているグループだが、グループ・マネージャ(308)によって後から定義されたグループをここで指定すること可能である。その場合はグループを指定する。2行目のような'-'のみの行は単なるセパレータである。3行目以降については、Type以下にならんでいる名前が、ユーザ操作名登録インタフェース(303)で登録された、ユーザ操作名である。また、各グループ名以下に書かれているのが、そのグループにおける問い合わせ処理の記述である。この記述は以下の3種類の構文のいずれかに当てはまる:

1. [問い合わせ処理の指定] ([問い合わせのタイムアウト]): [要求YESカウント]
2. [確認処理の指定] ([確認のタイムアウト])
3. [問い合わせ処理の指定] ([問い合わせのタイム

行する(S605)。

【0033】問い合わせ処理が指定されていた場合は(S602)、ユーザ問い合わせインタフェースプログラム(312)に、指定に基づき問い合わせパネル表示等の要求(314)を出し、問い合わせ先ユーザ(313)の返答を待つ(S603)。ユーザの問い合わせに対する返答は、要求(314)のリプライ(315)としてイベント・ハンドラ(305)に返される。ユーザ(313)の返答(315)の結果、ユーザ操作実行が可能ならば(S604)、本来のユーザ操作処理を直ちに実行する(S605)。逆に、ユーザが操作の実行を拒否する等により、ユーザ操作実行が不可能ならば(S604)、イベント・ハンドラ(305)はユーザ操作処理を行なうことなく終了する。

【0034】ここで、問い合わせ処理情報ファイル(310)に記述されてる情報について説明する。

【0035】以下に示すのは、問い合わせ処理情報ファイル(310)の内容の例である。

アウト]): [要求YESカウント] [[確認処理の指定] ([確認のタイムアウト])]

以下各構文の構成要素について説明する。

【0036】・問い合わせ処理の指定

問い合わせ処理には以下の指定が可能である。

【0037】X -- 問い合わせ処理を行なわない

N -- 問い合わせパネルを表示する。タイムアウト時はNOとして処理する

Y -- 問い合わせパネルを表示する。タイムアウト時はYESとして処理する

タイムアウトの時間は、[問い合わせのタイムアウト]秒である。タイムアウトが省略された場合はタイムアウト時間は無限大としてあつかう。

【0038】・確認処理の指定

まず、構文2における確認処理には以下の指定が可能である。

【0039】x -- 確認処理を行なわない

y -- 確認パネルを表示する

タイムアウトの時間は、[確認処理のタイムアウト]秒である。確認処理においてもタイムアウトが省略された場合はタイムアウト時間は無限大としてあつかう。また、構文3における確認処理の指定は、確認処理をおこなうべきグループ名を'&'でつないだものとなってい

る。これは、問い合わせ処理の結果をあらためてユーザに通知するといった形の処理に対応したものである。

【0040】・要求YES カウント

主に複数のユーザからなるグループに対する問い合わせで意味を持ち、最低どれだけの数のYES がそのグループ内で得られれば、全体としての回答がYES とみなされるかを示すもので、以下の数値あるいは文字列が指定される。

【0041】a. 0以上の整数

b. all (グループ内の全員)

c. majority (グループ内の半分+1)

d. aからb までを加減乗除演算子でくみあわせたもの
問い合わせ処理にのみ指定可能であるが不要な場合は省略される。

【0042】図4 は、図3 において説明した処理で、イベント・ハンドラ(305) が問い合わせ処理指定を含むリプライ・メッセージ(311) を受け取る直前の状況をあらわしている。

【0043】また図8 は図4 におけるイベント・ハンドラ(401) の処理をフローチャートであらわしたものである。

【0044】図4 において、イベント・ハンドラ(401) が問い合わせ処理指定(408) をvc_sm(402)内のダイアログ・マネージャ(403) から受けとったものとする。この問い合わせ処理指定(408) では、上記の問い合わせ処理情報ファイルの最初の1 行(これを定義行とよぶ)と、処理対象であるユーザ操作に対する問い合わせ処理が記述された行(これを記述行とよぶ)が文字列としてそのまま送られてきている。

【0045】イベント・ハンドラ(401) は問い合わせ処理指定(408) を受けとると、定義行および記述行の先頭から順に、以下の処理を繰り返し適用していく。

【0046】すなわち、グループ名と問い合わせ処理記述を問い合わせ処理指定(408) から取り出し(S801)、まず、グループ名を付加したグループ・ユーザ問い合わせ要求(409) をグループ・マネージャ(404) に発行する(S802)。すると、グループ・マネージャ(404) は、ハードディスク(406) 中のユーザ管理情報ファイル(407) から指定されたグループに属する全ユーザのリストを取りだし、それをリプライ・メッセージ(410) に付加して、イベント・ハンドラ(401) に返す。

【0047】イベント・ハンドラ(401) はユーザのリストを受けとると、リスト中の全ユーザのユーザ環境で稼働中のvc_yn(405)に、問い合わせ処理記述に対応するパネル表示要求(411)を発行する(S803)。このとき、図4 のように一般に問い合わせ処理の対象は複数のユーザが想定される。

【0048】仮に1 人であっても以下の処理は複数のユーザと同様に実行される。すなわち、イベント・ハンドラ(401) は、要求発行先すべてのvc_yn(405)からのリプ

ライ・メッセージ(412) が返ってくるのを待ち(S804)、その結果YES を返したユーザが問い合わせ処理記述中の〔要求YES カウント〕の値以上ならば(S805)全体としてYES が得られたとみなし、ユーザ操作の処理を続行する(S806)。〔要求YES カウント〕の値を下まわった場合は全体としてNOであったとして、ユーザ操作の処理は中止される(S807)。確認処理においても、すべてのvc_yn での確認処理が終了するまで待機するのは同じだが、この場合は最終的な結果には意味がないため、リプライ・メッセージ内のYES の個数の評価は特におこなわない点のみが異なっている。以上の操作が、定義行で指定された全グループに対してファイル中に記述された順番で適用される。

【0049】以上の様にして本実施例では、会議を抽象化したセッションにおける問い合わせ及び確認処理の対象となる操作を指定する手段と、

・セッションにおいて問い合わせ及び確認処理の対象となるユーザのグループを指定する手段と、

・操作とユーザの組み合わせに対する問い合わせ及び確認処理方式を指定する手段を問い合わせ処理情報ファイルとして有し、さらに、問い合わせ及び確認処理の実行手段としてvc_am, vc_yn等のソフトウェア群を提供している。

【0050】本実施例では、会議における問い合わせ処理、特に複数のユーザに対する問い合わせを柔軟に行うことが可能となる。

【0051】また、問い合わせ処理記述を通常のテキストファイルとして管理しているので、修正や変更に特別な機構が不要である。さらに、1 つの操作に対して、複数のグループへの問い合わせを指定することにより、段階的に問い合わせを行うことが可能となる。

【0052】(第2の実施例) 第1 の実施例においては、本発明をコンピュータ等の処理装置上の通信会議管理システムにおいて実施した場合について述べたが、第2の実施例としては、複数地点に配置されたカメラを制御することができるAwareness システムにおける実施について説明する。

【0053】このAwareness システムは、適当な位置に配置されたカメラの映像を適宜ディスプレイに呼びだし遠隔地の現在の状況を居ながらにして把握することを可能にするシステムである。会議などの厳密な意味での共同作業とは異なるが、共同作業の前段階としてのユーザ間の情報交換の手段を提供するという意味で、広義の共同作業システムとして本発明を適用することが可能である。まず、図9 を用いてこのAwareness システムの主要な構成要素について説明する。

【0054】図9 において、ワークステーション(901) は本実施例における処理の実行を行うCPU(902)と、本awareness システムの基本ソフトウェア群(911, 914, 918)を保持する主記憶(903)を計算機バス(905)で結合

することにより構成されている。また、本実施例における処理に必要な永続データを保持するためのデータベース(906)と、状況把握の手段となるコンピュータ制御可能なカメラ(907)およびユーザがシステムの操作をおこなうためのユーザインタフェースを表示するためのディスプレイ(904)がワークステーション(901)に結合されている。データベース(906)中には、認証処理情報(908)とポイント管理情報(909)およびユーザ管理情報(910)が置かれている。

【0055】データベース(906)自身は、一般的なりレ
10 ショナル・データベースやオブジェクト指向データベースなど、データの永続化と一貫性管理の機構をもっているものならどのようなものでも利用可能である。ここで、認証処理情報(908)は、本実施例で説明される認証処理を制御するために必要な情報である。

【0056】この認証処理とは、あるユーザがある地点の映像を見たり、カメラの制御など要求した場合に、その要求を実行して良いか否かを決定するための手続きのことである。また、ポイント管理情報(909)は、本実施例で説明される認証処理の対象となるカメラ(907)やその他のデバイスの管理のために必要な情報である。

【0057】この情報には、カメラの種類や遠隔操作の可否、置かれている場所などの情報が付加されている。ユーザ管理情報(910)は、本実施例で説明される認証処理の対象となるユーザおよびユーザのグループを管理するために必要な情報である。次に、本システムの基本ソフトウェア群について説明する。

【0058】・Awareness インタフェース(911)

Awareness インタフェース(911)は、状況把握のためのカメラ画像の表示やカメラ制御(上下左右の映像移動や
30 ズームなど)などのユーザインタフェースを提供するプログラムである。ユーザがこのAwareness インタフェース(911)を起動することにより、状況把握が可能となる。このAwareness インタフェース(911)には内部的に2つの重要なモジュールが存在する。

【0059】・ユーザ・マネージャ(912)

ユーザおよびユーザのグループに関する情報(ユーザ名、ユーザの現在の状態、グループのメンバなど)をデータベース(906)を利用して管理するモジュールである。

【0060】・ポイント・マネージャ(913)

カメラ等Awareness インタフェース(911)から利用可能なデバイスの管理情報であるポイント管理情報(909)をデータベース(906)を利用して管理するモジュールである。本実施例では、データベース(906)を利用しているため、実施例1におけるvc_smのような集中管理用の専用データ・サーバは不要となっている。

【0061】・デバイス制御サーバ(914)

Awareness インタフェースがユーザによって適宜起動されるのに対し、デバイス制御サーバ(914)はカメラ等利

用可能なデバイスが結合されているマシンで常に動作しているサーバである。図9においては、カメラ(907)が結合されているので、ワークステーション(901)上でデバイス制御サーバ(914)が動作している。デバイス制御サーバ(914)の機能は、Awareness インタフェースを介してユーザが行なった操作を実際に解釈し、実行することである。デバイス制御サーバ(914)は以下の2つの重要なモジュールを持っている。

【0062】・デバイス制御コマンド登録インタフェースモジュール(915)

第1の実施例におけるユーザ操作トリガマネージャ(113)と同様に、処理対象となるカメラ操作と、カメラコントロールモジュール(916)内におけるカメラ操作につけられた名前の関連づけを作成するための、プログラム・インタフェースを提供しているモジュールである。本実施例では、デバイス制御ためのコマンド群が、このモジュールによってカメラコントロールモジュール(916)内の認証処理対象コマンドのリストに登録される。

【0063】・カメラコントロールモジュール(916)

20 Awareness インタフェースなどからのカメラ操作要求を解釈・実行するモジュールである。本実施例のデバイス制御はカメラを主に対象としているため、カメラコントロールモジュールについて説明しているが、他のデバイスの制御も行う場合はそのデバイス用のコントロールモジュールがデバイス制御サーバ(914)に組み込まれる。このデバイスコントロール用モジュールの組み込みは、通常デバイス制御サーバ(914)の作成時に行われるが、動的リンクがサポートされているオペレーティングシステムが利用可能ならば、デバイス制御サーバ(914)の動作中に適宜モジュールを追加していくことも可能である。また、問い合わせパネルハンドラ(917)は第1の実施例における図2で示したような問い合わせパネルを表示させるためのモジュールである。

【0064】・認証サーバ(918)

本実施例において、Awareness インタフェース(911)等からおくられてきたデバイス制御コマンドをデバイス制御サーバ(914)で実行可能か否かを決定するサーバである。認証サーバ(918)は以下のモジュールを内蔵している。

40 【0065】・認証情報マネージャ(919)

Awareness インタフェース(911)からカメラ等利用可能なデバイスを利用する際の認証に必要な認証処理情報(908)をデータベース(906)を利用して管理するモジュールである。認証サーバ(918)は、認証情報マネージャ(919)が管理している認証処理情報(908)に基づいて、カメラ等デバイス操作の可否を決定する。ここでは、まず認証処理情報(908)の内容について説明する。認証処理情報(908)には以下の項目が存在する。

【0066】Command: {コマンド名}

50 Free-Users: {無条件で利用可能なユーザのリスト}

Possible-Users: [問い合わせで許可されるならば利用可能なユーザのリスト]

これらが、認証処理の対象になる全コマンドについて定義される。ここでユーザのリストとは、ユーザ名を','で区切って並べたものである。[無条件で利用可能なユーザのリスト]に登録されているユーザは[コマンド名]で指定される操作を無条件で許される。また、[無条件で利用可能なユーザのリスト]にも[問い合わせで許可されるならば利用可能なユーザのリスト]にも属さないユーザは指定された操作を行うことができないことになる。

【0067】一方[問い合わせで許可されるならば利用可能なユーザのリスト]に属するユーザに対しては、実施例1と同様な問い合わせ処理の指定が必要となる。すなわち、各ユーザに対して

[ユーザ名]: [問い合わせ対象ユーザのリスト]:
[要求YES カウント]

が指定され、[問い合わせ対象ユーザのリスト]の許可を求めることになる。実際には、問い合わせ対象ユーザの環境で動作しているデバイス制御サーバが問い合わせ用のパネルを表示する。ここで、[問い合わせ対象ユーザのリスト]中のユーザ名に'([タイムアウト時間])'を付加することにより、ユーザごとにタイムアウト時間を指定できる。尚、指定しなかった場合のデフォルト値は20秒である。また、[要求YES カウント]は、実施例1とおなじく、

- a. 0以上の整数
- b. all (グループ内の全員)
- c. majority (グループ内の半分+1)
- d. aからbまでを加減乗除演算子でくみあわせたものを指定可能である。

【0068】以上の様にして定義された認証処理情報の例は以下ようになる。'#'で始まる行はコメントとして無視される。

【0069】# ヘッダ情報

```
Command:      SoomUpCamera
Free-Users:    fukasawa, okasaki, sato
Possible-Users: kawazome, ono
# 制限つきユーザの記述
```

```
kawazome:      sato(10), bannai(20): all
ono:           bannai(20)
```

では、図10と図11を用いて本システムにおける処理の流れについて説明する。このうち図10は処理の流れの概念図であり、図11は図10における認証情報マネージャ(101)の動作をフローチャートであらわしたものである。

【0070】図10において、1人のユーザ(1001)がAwareness インタフェース(1002)を起動してユーザ(1005)の管理下にあるデバイス制御サーバ(1004)の管理下にあるカメラ(1006)の映像の参照を要求したと仮定する。

【0071】この映像取得要求(1014)には、ユーザ・マ

ネージャ(1004)が管理しているユーザ情報とポイント・マネージャ(1003)で管理されていたデバイス(ここではカメラ)情報が付加される。これらの情報は、Awareness インタフェース(1002)起動時に自動的に、データベース(1012)から取り出される。映像取得要求(1014)が、デバイス制御コマンドインタフェースモジュール(1008)によってデバイス制御モジュールに登録されたものならば、デバイス制御モジュール(1009)はその操作の認証要求(1015)を認証サーバ(1010)に発行する。

【0072】この認証要求(1015)には映像取得要求(1014)に付加されていたユーザ情報とデバイス情報がそのまま付加される。

【0073】さて、認証サーバ(1010)が認証要求(1015)を受けると、認証情報マネージャ(1011)で処理される。すなわち、認証情報マネージャ(1011)は、データベース(1012)から認証処理情報(1013)をとりだし(S1101)、もし要求したユーザが制限なしユーザなら(S1102)、直ちに処理の続行をリプライ・メッセージ(1017)でデバイス制御サーバ(1007)に通知する(S1103)。この場合デバイス制御サーバ(1007)はカメラ制御の認証に成功したとみなして処理を続行する。

【0074】また、要求したユーザが制限なしユーザでも制限ありユーザでもなかった場合は(S1102, S1104)、同様にして直ちに処理の中止をデバイス制御サーバ(1007)に通知し、デバイス制御サーバ(1007)は処理を中止する(S1105)。ユーザが制限ありユーザであった場合、認証情報マネージャ(1011)はさらに認証処理情報(1013)内を走査し、ユーザ1001に関する問い合わせ記述を抜き出す(S1106)。そして、その中の[問い合わせ対象ユーザのリスト]と[要求YES カウント]をとりだし(S1107, S1108)、リスト中の全ユーザに対して問い合わせパネル表示を行なわせる(S1109)。

【0075】問い合わせパネル表示は、対象ユーザのユーザ環境で動作しているデバイス制御サーバ内の問い合わせパネル・ハンドラ(1009)によって行なわれる。ここで、対象ユーザの環境でデバイス制御サーバが起動されていなかった場合は(S1110)、そのユーザは一時的にリストから除外されて処理が行なわれる(S1111)。

【0076】図10においては、デバイス制御サーバ(1007)の問い合わせパネル・ハンドラ(1009)にパネル表示要求(1018)が発行され、ユーザ(1005)に対して問い合わせが行われている。認証情報マネージャ(1011)は除外されなかったユーザからの返答を待ち(S1112)、その結果

[要求YES カウント]分の同意が得られたなら(S1113)、直ちに処理の続行をデバイス制御サーバ(1007)に通知する(S1115)。この場合デバイス制御サーバ(1007)はカメラ制御処理を行うことが可能となる。逆に[要求YES カウント]分の同意が得られなかったなら(S1113)、直ちに処理の中止をデバイス制御サーバ(1007)に通知する(S1114)。この場合デバイス制御サーバ(1007)はカメ

ラ制御処理を行うことはできない。以上の様にして本実施例では、

- ・ Awareness システムにおける認証およびそれともなう問い合わせ処理の対象となる操作を指定する手段と、
- ・ Awareness システムにおける認証および問い合わせ処理の対象となるユーザのグループを指定する手段と、
- ・ 操作とユーザの組み合わせに対する問い合わせ処理方式を問い合わせ処理情報記述という形で提供し、さらに、認証および問い合わせ処理の実行手段として認証サーバ、デバイス制御サーバ等のソフトウェア群を提供した。

【0077】本実施例では本発明を適用することにより、Awareness システムにおける認証に関する処理とデバイスへのアクセス権に関する処理を統合し統一的に扱うことが可能となる。

【0078】また、Awareness システムで利用している情報やメカニズムを、他の共同作業システムと共有することが可能となるため、Awareness システムと他の共同作業システムとの統合が容易となる。

【0079】

【発明の効果】以上の様にして、共同作業を行う処理システムにおいて、

- ・ 共同作業上で問い合わせが必要な操作を指定するための手段と
- ・ 指定された操作において、意味づけされるユーザおよびユーザのグループを指定する手段を提供する。

さらに、上記操作とユーザ、グループの組み合わせに対応する問い合わせ処理を指定する手段を提供し、その指定手段によって指定された問い合わせ処理の実行手段を

提供することにより

- (1) 状況に応じて変化しうる複雑な共同作業における問い合わせ処理機構を提供することができる。
 - (2) 任意の共同作業における最適な問い合わせ処理の形態を提供することが容易となる。
- という、効果を得ることができるようになる。

【図面の簡単な説明】

【図1】本発明の第1の実施例である会議管理システムの構成図である。

【図2】第1の実施例における問い合わせ処理のためのユーザインタフェースの図である。

【図3】第1の実施例における基本的なユーザ操作問い合わせの処理の流れを説明するための図である。

【図4】イベント・ハンドラによる問い合わせ記述の解釈・実行の流れを説明するための図である。

【図5】図3で示す処理における会議操作インタフェースプログラムの動作をあらわすフローチャートである。

【図6】図3で示す処理におけるイベント・ハンドラの動作をあらわすフローチャートである。

【図7】図3で示す処理におけるダイアログ・マネージャの動作をあらわすフローチャートである。

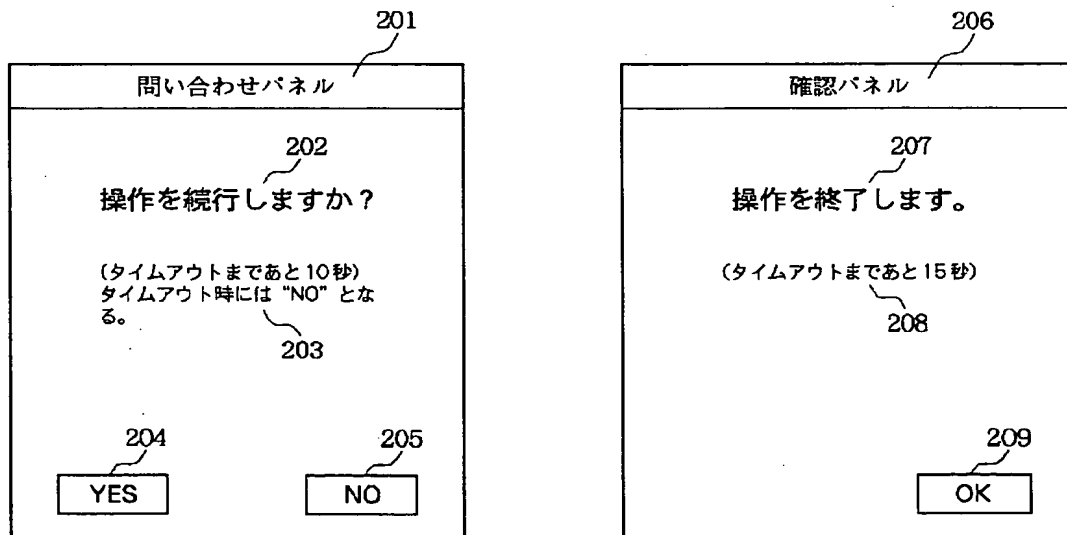
【図8】図4で示す処理におけるイベント・ハンドラの動作をあらわすフローチャートである。

【図9】本発明の第2の実施例である Awareness (状況把握) システムの構成図である。

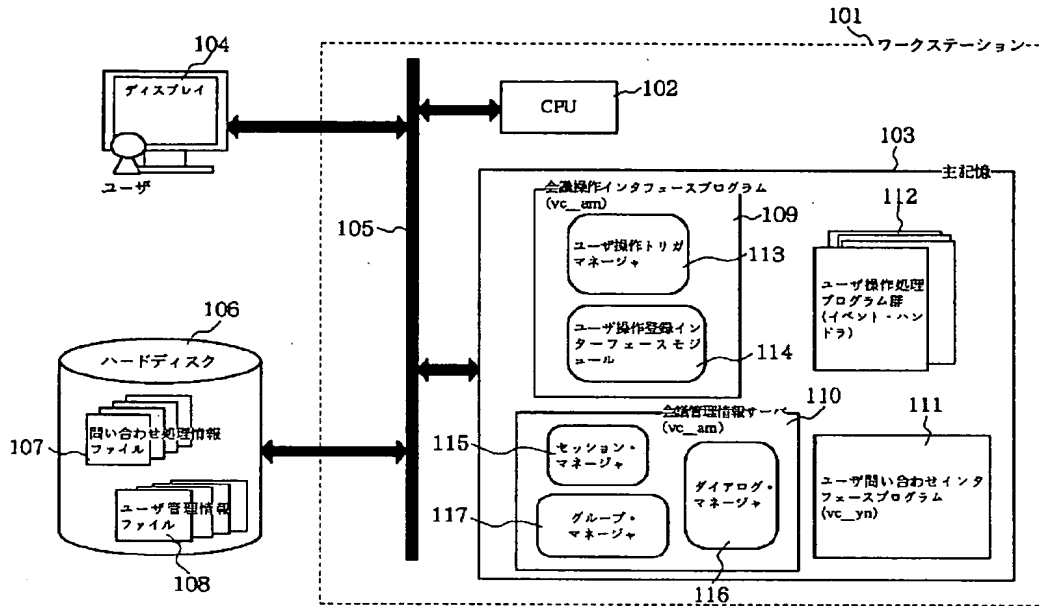
【図10】第2の実施例における基本的なユーザ操作問い合わせの処理の流れを説明するための図である。

【図11】図10で示す処理における認証情報マネージャの動作をあらわすフローチャートである。

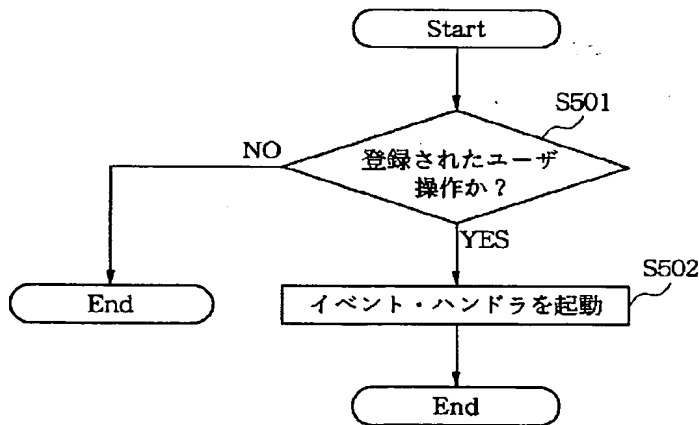
【図2】



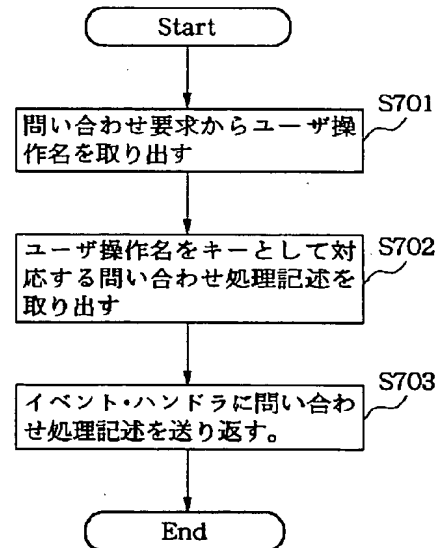
【図1】



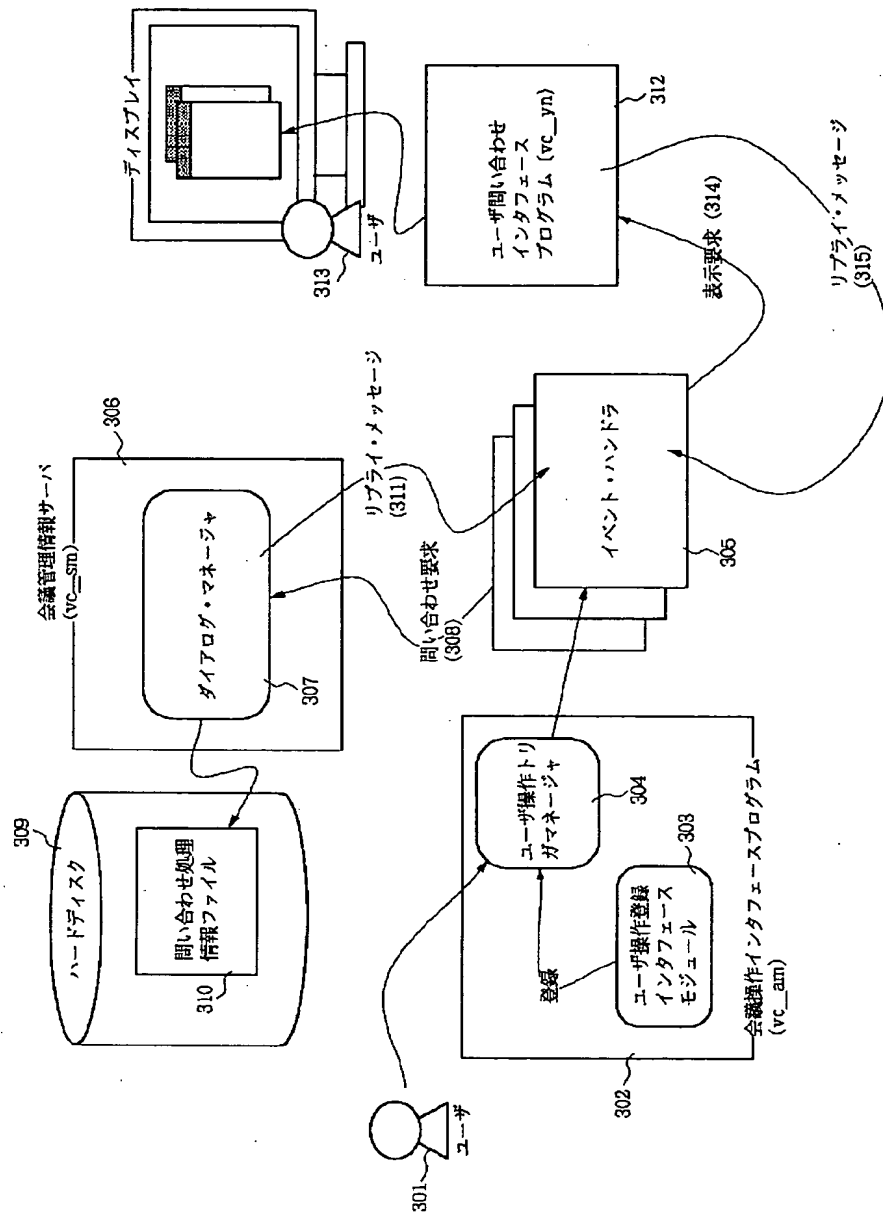
【図5】



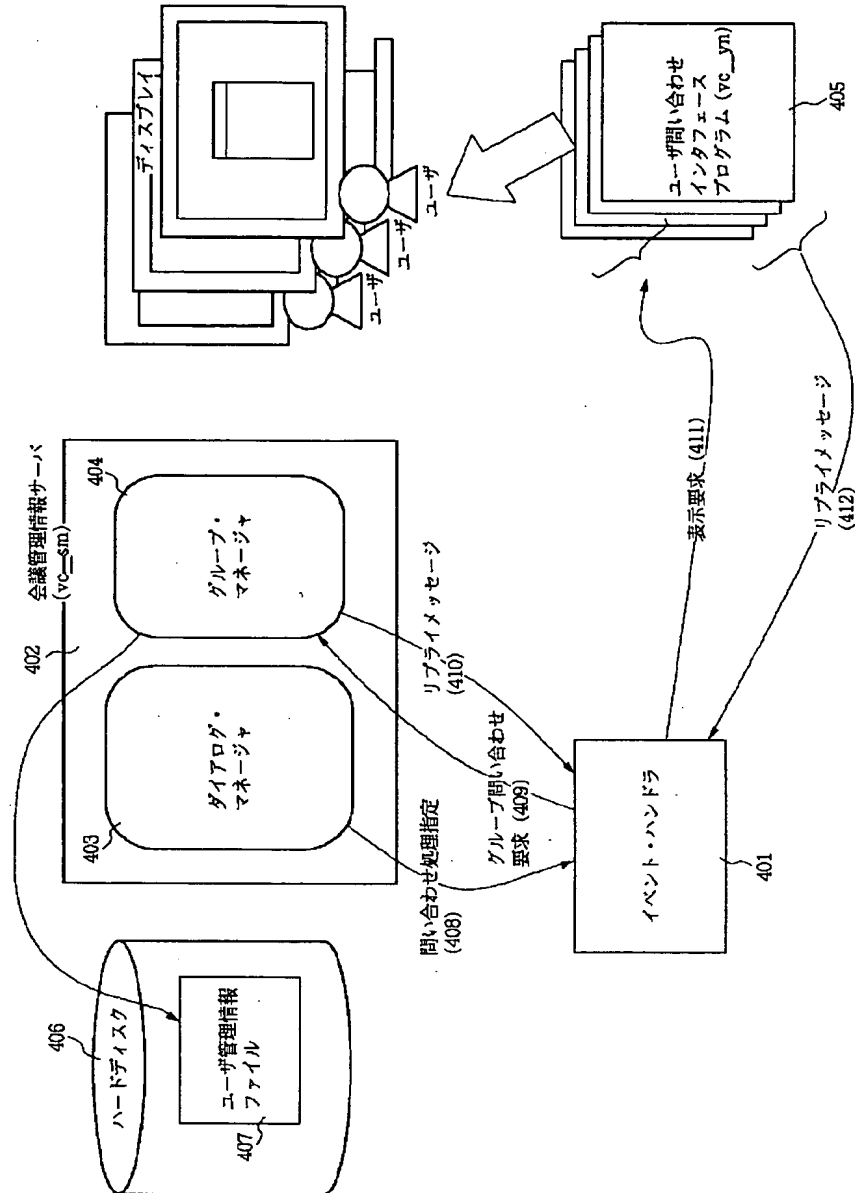
【図7】



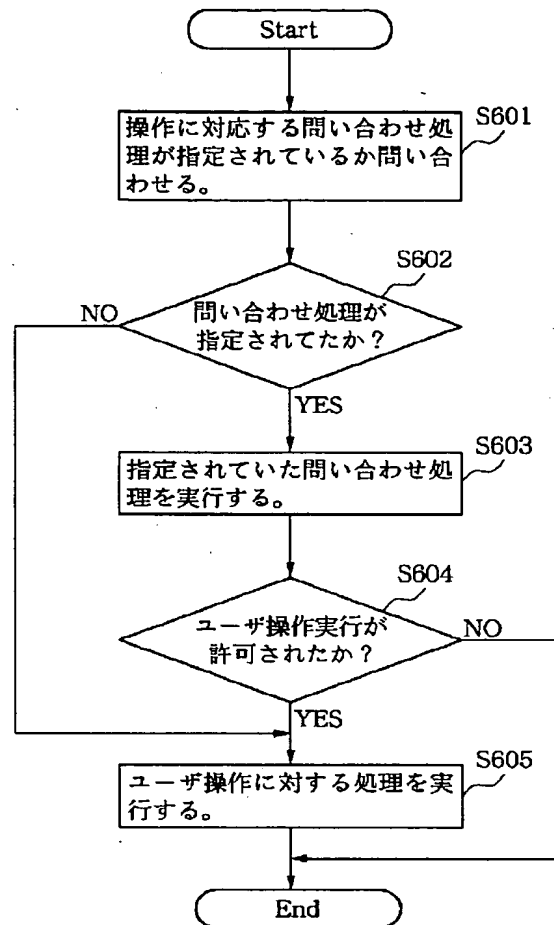
【図 3】



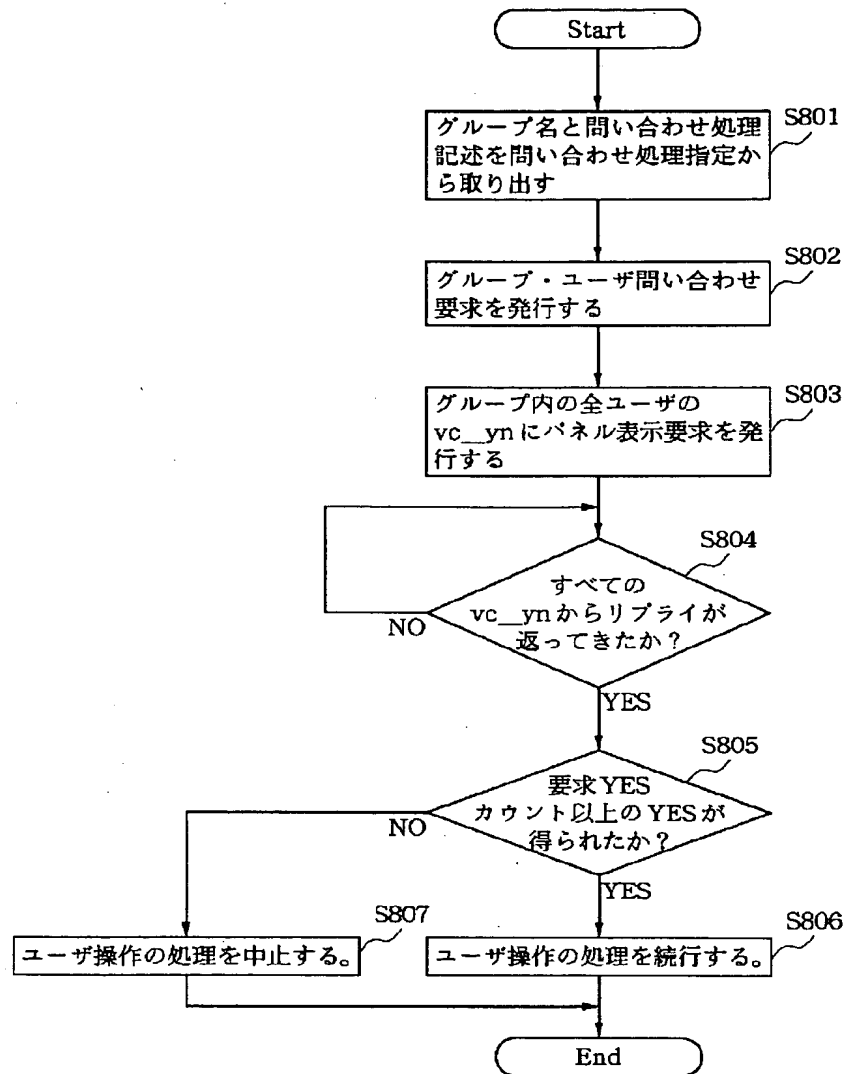
【図 4】



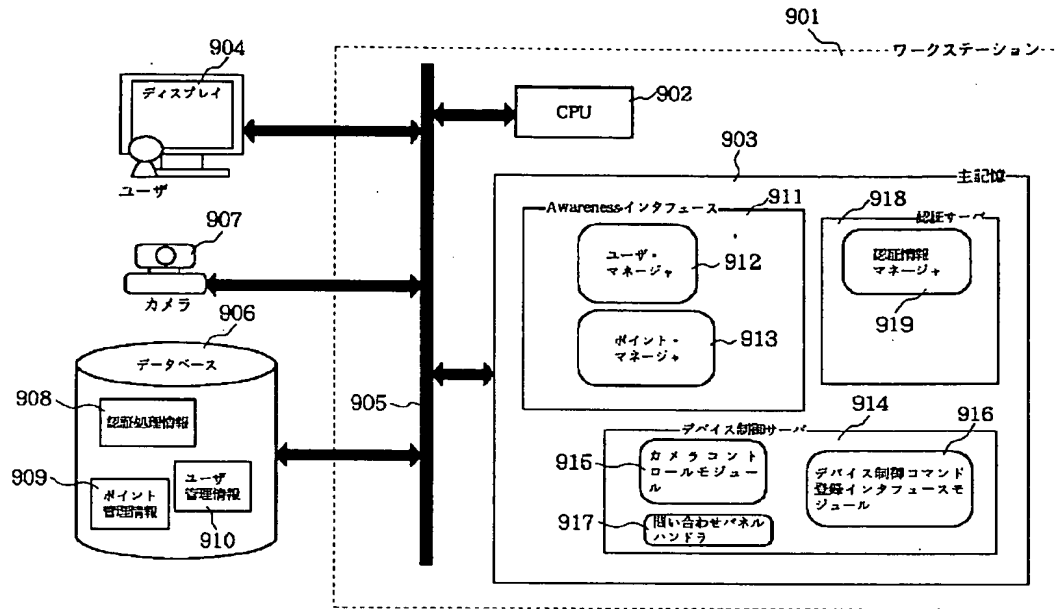
【図 6】



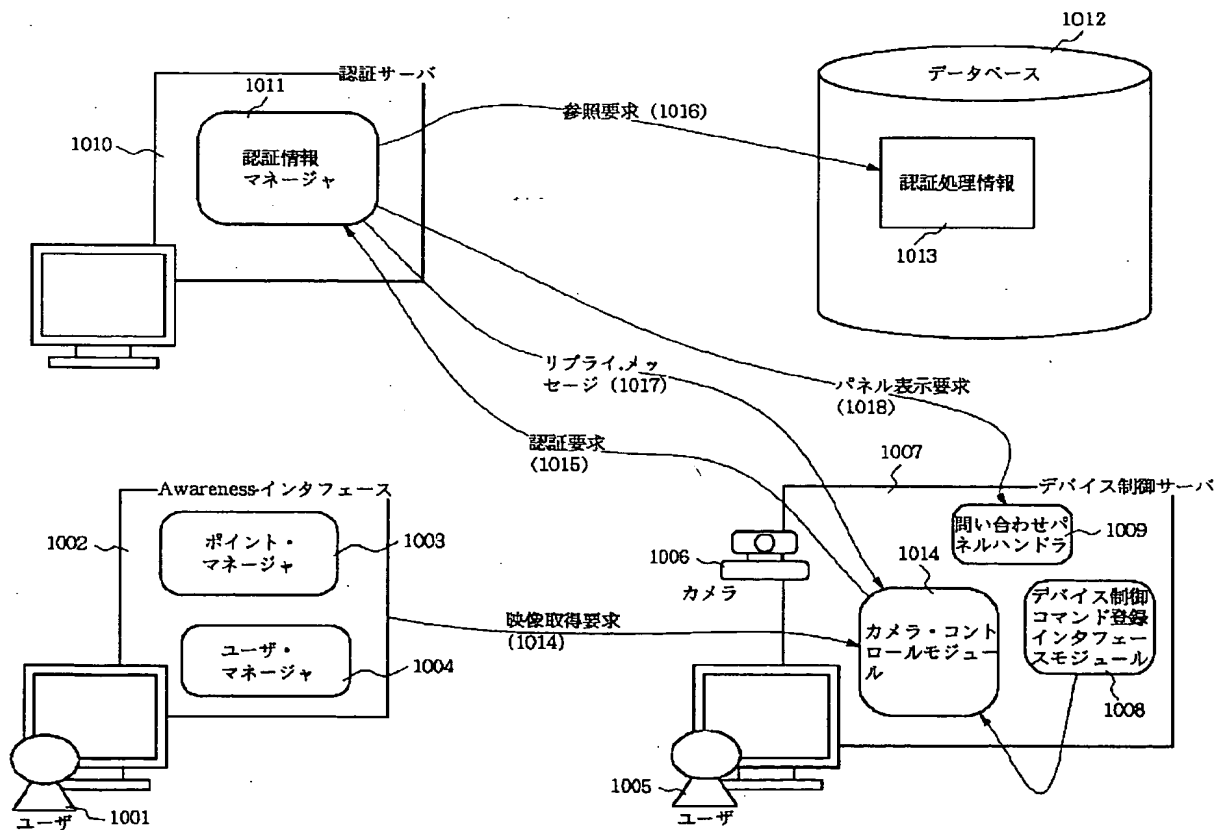
【図 8】



【図 9】



【図 10】



【図 11】

